WE CLAIM

1. Apparatus for processing data, said apparatus comprising:

(i) a processor core having a register bank containing a plurality of registers and being operable to execute operations upon register operands held in said registers as specified within instructions of a first instruction set; and

(ii) an instruction translator operable to translate instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set, instructions of said second instruction set specifying operations to be executed upon stack operands held in a stack; wherein

(iii) said instruction translator is operable to allocate a set of registers within said register bank to hold stack operands from a portion of said stack;

(iv) said instruction translator has a plurality of mapping states in which different registers within said set of registers hold respective stack operands from different positions within said portion of said stack; and

(v) said instruction translator is operable to change between mapping states in dependence upon operations that add or remove stack operands held within said set of registers.

2. Apparatus as claimed in claim 1, wherein said translator output signals include signals forming an instruction of said first instruction set.

3. Apparatus as claimed in any one of claims 1 and 2, wherein said translator output signals include control signals that control operation of said processor core and match control signals produced on decoding instructions of said first instruction set.

4. Apparatus as claimed in any one of claims 1, 2 and 3, wherein said translator output signals include control signals that control operation of said processor core and specify parameters not specified by control signals produced on decoding instructions of said first instruction set.

5. Apparatus as claimed in any one of the preceding claims, wherein said instruction translator provides mapping states such that stack operands are added to or removed from said set of registers without moving stack operands between registers within said set of registers.

6. Apparatus as claimed in any one of the preceding claims, wherein said set of registers are operable to hold stack operands from a top portion of said stack including a top of stack operand from a top position within said stack.

7. Apparatus as claimed in any one of the preceding claims, wherein said stack further comprises a plurality of addressable memory locations holding stack operands.

8. Apparatus as claimed in claim 7, wherein stack operands overflow from said set of registers into said plurality of addressable memory locations.

9. Apparatus as claimed in any one of claims 7 and 8, wherein stack operands held within said plurality of addressable memory locations are loaded into said set of registers prior to use.

10. Apparatus as claimed in any one of the preceding claims, wherein said instruction translator uses a plurality of instruction templates for translating instructions from said second instruction set to instructions from said first instruction set.

11. Apparatus as claimed in claim 10, wherein an instruction from said second instruction set including one or more stack operands has an instruction template comprising one or more instructions from said first instruction set in which register operands are mapped to said stack operands.

12. Apparatus as claimed in any one of the preceding claims, wherein said instruction translator comprises one or more of:

(i) hardware translation logic;

(ii) instruction interpreting program code controlling a computer apparatus;

(iii) instruction compiling program code controlling a computer apparatus; and

(iv) hardware compiling logic.

13. Apparatus as claimed in any one of the preceding claims, wherein said instruction translator includes a first plurality of state bits indicative of a number of stack operands held within said set of registers.

14. Apparatus as claimed in claim 6 and any one of claims 5 and 7 to 13, wherein said instruction translator includes a second plurality of state bits indicative of which register within said set of registers holds said top of stack operand.

15. Apparatus as claimed in any one of the preceding claims, wherein said second instruction set is a Java Virtual Machine instruction set.

16. A method of processing data using a processor core having a register bank containing a plurality of registers and being operable to execute operations upon register operands held in said registers as specified within instructions of a first instruction set, said method comprising the steps of:

(i) translating instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set, instructions of said second instruction set specifying operations to be executed upon stack operands held in a stack;

(ii) allocating a set of registers within said register bank to hold stack operands from a portion of said stack;

(iii) adopting one of a plurality of mapping states in which different registers within said set of registers hold respective stack operands from different positions within said portion of said stack; and

(iv) changing between mapping states in dependence upon operations that add or remove stack operands held within said set of registers.

17. A computer program product holding a computer program for controlling a computer to perform the method of claim 16.

18. Apparatus for data processing substantially as hereinbefore described with reference to the accompanying drawings.

19. A method of data processing substantially as hereinbefore described with reference to the accompanying drawings.

20. A computer program product holding a computer program for controlling a computer to perform a method substantially as hereinbefore described with reference to the accompanying drawings.